



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/689,186	10/20/2003	Sheng Ye	CN920020009US1	2701
75234 7590 04/01/2009 RYAN, MASON & LEWIS, LLP 90 FOREST AVENUE LOCUST VALLEY, NY 11560			EXAMINER HILLERY, NATHAN	
			ART UNIT 2176	PAPER NUMBER
			MAIL DATE 04/01/2009	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/689,186
Filing Date: October 20, 2003
Appellant(s): YE ET AL.

David E Shifren
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 1/13/09 appealing from the Office action
mailed 6/13/08.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Barr, Michael; Programming Embedded Systems in C and C++; Publisher: O'Reilly;
Pub Date: January 1999; Pages: 1 and 2

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 23 – 42 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Specifically, the specifics can not be ascertained from the specification nor the drawings regarding the following limitations:

- a. transferring the at least one data unit from the one or more locations within the at least one input file to the one or more locations within the first output file specified by a mapping of the at least one data unit of the at least one input file to one or more locations within the first output file;
- b. wherein each location comprises a horizontal position, the horizontal position comprising at least one of an uppermost position of the data unit or a lowermost position of the data unit, and a vertical position, the vertical position comprising at least one of the leftmost position of the data unit or the rightmost position of the data unit; and

Art Unit: 2176

c. wherein each data unit is defined based on at least one of: at least one string, at least one absolute position of the data unit within the input file, at least one relative position of the data unit to a start or end of at least one of a row or column of the input file, and at least one relative position of the data unit to another data unit.

Further, claims 23 – 42 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Again, the specifics about the above mentioned limitations can not be ascertained from the specification nor the drawings. It is not understood how the claimed invention is suppose to operate even in light of the disclosure, thus, effecting the interpretation of the claims under 35 USC 102(b).

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 23 – 42 are rejected under 35 U.S.C. 102(b) as being anticipated by Barr (Programming Embedded Systems in C and C++).

Regarding independent claim 23,

Art Unit: 2176

Barr teaches that a symbol table somewhere in the object file that contains the names and locations of all the variables and functions referenced within the source file (p 2, third full paragraph), which meet the limitation of **defining at least one data unit of the at least one input file; determining one or more locations within the at least one input file of the at least one data unit;**

Barr teaches that the contents of an object file can be thought of as a very large, flexible data structure. The structure of the file is usually defined by a standard format. If you'll be using more than one compiler (i.e., you'll be writing parts of your program in different source languages), you need to make sure that each is capable of producing object files in the same format (p 2, first full paragraph), which meet the limitation of **transferring the at least one data unit from the one or more locations within the at least one input file to the one or more locations within the first output file specified by a mapping of the at least one data unit of the at least one input file to one or more locations within the first output file;**

Barr teaches that a symbol table somewhere in the object file that contains the names and locations of all the variables and functions referenced within the source file (p 2, third full paragraph), which meet the limitation of **wherein each location comprises a horizontal position, the horizontal position comprising at least one of an uppermost position of the data unit or a lowermost position of the data unit, and a vertical position, the vertical position comprising at least one of the leftmost position of the data unit or the rightmost position of the data unit.**

Art Unit: 2176

Barr teaches that each of these sections contains one or more blocks of code or data that originated within the original source file (p 2, second full paragraph), which meet the limitation of **wherein each data unit is defined based on at least one of: at least one string, at least one absolute position of the data unit within the input file, at least one relative position of the data unit to a start or end of at least one of a row or column of the input file, and at least one relative position of the data unit to another data unit;**

Regarding dependent claims 24 and 25, Barr teaches that all of the code blocks are collected into a section called text (p 2, second full paragraph), which meet the limitation of **the at least one string is within the data unit and the at least one string is adjacent to the data unit.**

Regarding dependent claim 26, Barr teaches that regardless of the input language (C/C++, assembly, or any other), the output of the cross-compiler will be an object file. This is a specially formatted binary file that contains the set of instructions and data resulting from the language translation process (p 1, fourth paragraph), which meet the limitation of **the step of transferring the at least one data unit comprises transforming the at least one data unit from a first format to a second format.**

Regarding dependent claim 27, Barr teaches that the contents of an object file can be thought of as a very large, flexible data structure. The structure of the file is

Art Unit: 2176

usually defined by a standard format. If you'll be using more than one compiler (i.e., you'll be writing parts of your program in different source languages), you need to make sure that each is capable of producing object files in the same format (p 2, first full paragraph), which meet the limitation of **further comprising a step of generating a second output file from the at least one input file by transferring the at least one instance of the data unit from the one or more locations within the at least one input file to the one or more locations within the second output file specified by the mapping of the at least one data unit of the at least one input file to one or more locations within the first output file.**

Regarding dependent claim 28, Barr teaches that the contents of an object file can be thought of as a very large, flexible data structure. The structure of the file is usually defined by a standard format. If you'll be using more than one compiler (i.e., you'll be writing parts of your program in different source languages), you need to make sure that each is capable of producing object files in the same format (p 2, first full paragraph), which meet the limitation of **further comprising a step of generating a second output file from the at least one input file by transferring the at least one instance of the data unit from the one or more locations within the at least one input file to one or more locations within the new output file specified by a mapping of the at least one data unit of the at least one input file to one or more locations within the second output file.**

Regarding dependent claim 29, Barr teaches that the contents of an object file can be thought of as a very large, flexible data structure. The structure of the file is usually defined by a standard format. If you'll be using more than one compiler (i.e., you'll be writing parts of your program in different source languages), you need to make sure that each is capable of producing object files in the same format (p 2, first full paragraph), which meet the limitation of **further comprising the step of generating a second output file from at least one new file by determining one or more locations within the at least another new file of the at least one data unit of the at least one input file and transferring the at least one data unit from the determined one or more locations within the at least one new file to the one or more locations within the output file specified one or more locations within the new output file specified by the mapping of the at least one data unit of the at least one input file to one or more locations within the first output file.**

Regarding claims 30 - 42, the claims incorporate substantially similar subject matter as claims 23 – 29 and are rejected along the same rationale.

(10) Response to Argument

Applicant argues the rejection made under 35 USC 112, first paragraph (pp 6 – 12).

The Office disagrees.

First, Limitation 'A' recites **transferring the at least one data unit from the one or more locations within the at least one input file to the one or more locations within the first output file specified by a mapping of the at least one data unit of the at least one input file to one or more locations within the first output file;**

Specifically, limitation 'A' recites the transference of data units from one location within the input file to another location within the output file specified by a mapping of the locations. The accompanying support cited by the applicant discloses transforming data in an original file into data in an objective file. Specifically, each of the data units is mapped to a corresponding format. Although the data units are 'located', the locations of the data units are not mapped so as to 'relocate' them to another file as now required by the claim. The skilled artisan would understand the disclosure to mean that the data units are simply being transformed into a different file format, which the skilled artisan would differentiate from 'relocating' data units in one file to another. These are two completely different processes, which have two completely different utilities.

Not only does the level of detail required by the claim not rise to the level disclosed in the specification but the concepts disclosed in the specification are different from the concepts required of the claim. Thus, the specification does not enable one to make and/or use the claimed invention.

Art Unit: 2176

Next, limitation 'B' recites that each location comprises a horizontal position and a vertical position in which the horizontal position has an uppermost position or lowermost position and the vertical position has a left most position or a right most position. Again, the accompanying support cited by applicant discloses that it takes four types of location elements to determine a single position of a data unit. It appears yet again that applicant has the benefit of hindsight and some years when drafting the amended claims that were not apparent or available at the time of the originally filed specification.

Lastly, limitation 'C' recites that each data unit is defined based on a number of options including a string. The cited portions of the specification, however, disclose that data units mainly consist of five types - text, single line, multi line, block, and iterator.

Applicant appears to make the biggest leap in this comparison. Further, applicant seems to argue that because the specification discloses that the invention should not be limited to the above five types of data units and that any data units for data locating may flexibly be incorporated when needed; then, the applicant can claim any type of data unit with any limitation.

Applicant generally argues the limitations 'B' and 'C' as they are rejected under 35 USC 102.

The Office disagrees.

It should first be noted that absent enablement and guidance from the specification, the office was forced to rely on the knowledge generally known to one of ordinary skill in the art in rejecting the claim limitations especially A, B, and C. To that end, the Office maintains that the reference anticipates the claim limitations with applicable sections cited in the final rejection in such much as could be understood in light of the rejections under 35 USC 112.

Secondly, limitations 'B' and 'C' amount to wherein clauses in a method claim that consists solely of **nonfunctional descriptive material**. The language of independent claim 23 does not require a teaching or suggestion in the prior art since all of the "wherein..." limitations recited in the claim are directed to non-functional descriptive material (the content) and do not functionally limit the claimed "method for generating..." The teachings of Barr clearly teach and fairly suggest the three active steps recited in the claim. The "wherein" limitations attempt to limit the data and do not change the method of generating an output file from an input file. Here, the active methods steps are the same irrespective of content or temporal relationship of the data. Also, applicants do not contest the functionality of the operation of Barr to generate at least a first output file from at least one input file.

Simply with respect to the "wherein..." limitations, these limitations are non descriptive material, which do not exhibit a functional relationship with a substrate and therefore do

Art Unit: 2176

not affect the manner in which the computer processes are performed. The nonfunctional descriptive material is treated as analogous to printed matter cases where what is printed on a substrate bears no functional relationship to the substrate and is given no patentable weight. See *In re Gulack*, 703 F.2d 1381, 1385 (Fed. Cir. 1983). See also *Ex parte Curry*, 84 USPQ2d 1272 (BPAI 2005) (nonprecedential) (Federal Circuit Appeal No. 2006-1003, *aff'd* Rule 36 Jun. 12, 2006). The Examiner need not give patentable weight to descriptive material absent a new and unobvious functional relationship between the descriptive material and the substrate. See *In re Lowry*, 32 F.3d 1579, 1582-83 (Fed. Cir. 1994); *In re Ngai*, 367 F.3d 1336, 1338 (Fed. Cir. 2004). See also *Ex parte Nehls*, <http://www.uspto.gov/web/offices/dcom/bpai/prec/fdO71823.pdf> (BPAI Jan. 28, 2008); *Ex parte Mathias*, 84 USPQ2d 1276 (BPAI 2005) (nonprecedential) (191 Fed. Appx. 959 (Fed. Cir. 2006)).

The alleged distinctions are based on nonfunctional descriptive material alone where no different functionality of process is imparted by the characterization of data as set forth respectively in representative independent claim 23.

MPEP 2111.04 states that claim scope is not limited by claim language that suggests or makes optional but does not require steps to be performed, or by claim language that does not limit a claim to a particular structure, and that the court noted (quoting *Minton v. Nat'l Ass'n of Securities Dealers, Inc.*, 336 F.3d 1373, 1381, 67 USPQ2d 1614, 1620 (Fed. Cir. 2003)) that a "whereby clause in a method claim is not given weight when it simply expresses the intended result of a process step positively recited." *Id.*

These limitations simply describe what the particular data is or should be prior to and/or after the positive method steps are performed. Thus, it is not given patentable weight.

However, for completeness, Barr teaches that a symbol table somewhere in the object file that contains the names and locations of all the variables and functions referenced within the source file (p 2, third full paragraph) and that each of these sections contains one or more blocks of code or data that originated within the original source file (p 2, second full paragraph), which meet the limitations under 35 USC 102 in light of the rejection under 35 USC 112, first paragraph.

Appellants argue that the relied-upon portion of Barr is clearly directed to generating different object files from different input files written in different source languages, rather than generating a plurality of output files from the same at least one input file, as recited in claims 27 and 28 (pp 14 and 15).

The Office disagrees.

First, Applicant mischaracterizes the reference. Barr teaches that the contents of an object file can be thought of as a very large, flexible data structure. The structure of the file is usually defined by a standard format. If you'll be using more than one compiler

Art Unit: 2176

(i.e., you'll be writing parts of your program in different source languages), you need to make sure that each is capable of producing object files in the same format (p 2, first full paragraph).

Applicant argues that the claim recites generating a plurality of output files from the same at least one input file and that Barr is directed to generating different object files from different input files written in different source languages. In actuality, Barr teaches generating plural object files from one input file or program, which has 'parts' written in different languages. Therefore, Barr in fact meets the claims' limitations.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Nathan Hillery/

Examiner, Art Unit 2176

Conferees:

/DOUG HUTTON/

Supervisory Patent Examiner, Art Unit 2176

/Stephen S. Hong/

Supervisory Patent Examiner, Art Unit 2178